ELSEVIER
Available online at www.sciencedirect.com
SciVerse ScienceDirect
Procedia Engineering 29 (2012) 4117 – 4123
Procedia Engineering
www.elsevier.com/locate/procedia

2012 International Workshop on Information and Electronics Engineering (IWIEE)

Improved IMM Algorithm for Nonlinear Maneuvering Target Tracking

Liang Gao[a] , Jianping Xing [a]* , Zhenliang Ma[a] , Junchen Sha[a], Xiangzhan Meng[a]

[a] *School of information science and engineering, Shandong University, 250100 Jinan, China*

Abstract

Devoted to the problem of state estimation of discrete-time stochastic systems, SIMM (Scalar-Weight Interacting Multiple Model) and MIMM (Matrix-Weight Interacting Multiple Model) methods are proposed by X. Fu, in which the filter outputs are combined based on two optimal multi-model fusion criterions weighted by scalars and general matrices, respectively. In this paper, four improved IMM algorithms (EKF-SIMM, EKF-MIMM, UKF-SIMM and UKF-MIMM) are presented for nonlinear maneuvering target tracking based on SIMM and MIMM. The proposed improved algorithms can receive the optimal state estimations of target in the nonlinear minimum variance sense. Experiments results verify the effectiveness of the proposed algorithms by comparing with EKF-IMM and UKF-IMM. And the proposed algorithms have an absolute advantage in the velocity estimation. In particular, UKF-MIMM is obviously better than EKF-IMM and UKF-IMM in accuracy while EKF-SIMM is superior in elapsed time. Therefore, the proposed algorithms can be competitive alternatives to the classical IMM–based filter algorithms for nonlinear maneuvering target tracking.

Keywords: nonlinear maneuvering target tracking, interacting multiple model, Unscented Kalman Filter, Extended Kalman Filter

1. Introduction

The state estimation problem of discrete-time stochastic systems with Markov switching parameters is always the focus of interest in the community of maneuvering target tracking. The Interacting Multiple Model (IMM) algorithm is a widely accepted scheme for solving this problem, which is generally nonlinear. The IMM algorithm has confusions of probability density functions (PDFs) and probability masses that denote the continuous-valued and discrete-valued parameter of stochastic process respectively,

* * *Corresponding author. Tel.: 13708921161.*
E-mail address: xingjp@sdu.edu.cn.

doi:10.1016/j.proeng.2012.01.630

which usually has been neglected [1]. In the process of IMM algorithm, updating weights of models are derived from the mixture of PDFs and probability masses, however, probability mass must be a value in the interval [0, 1], and any PDF has no such restriction. Thus, the two kinds of values are at different levels that lead to a certain error in many cases.

There have been many related researches to improve the IMM algorithm. For example, Johnstona.L et.al [2] derived a weighted IMM with a recursive implementation of maximum posteriori (MAP) state sequence estimator; Hong, L. et.al [3] proposed a multi-rate interacting multiple model particle filter (MRIMM-PF) to effectively solve the problem of nonlinear and non-Gaussian tracking problem, with an emphasis on computational savings. But these improvements can't be devoted to the problem that the confusion of probability density functions (PDFS) and probability masses in IMM algorithm. Two improved IMM algorithms [1], using multi-model fusion criterions weighted by scalars and matrices, respectively, together with the step of calculating the mixed initial model weights, state and corresponding covariance, named scalar-weight IMM (SIMM), matrix-weight IMM (MIMM), were proposed to solve the above problem. The two algorithms have a better Performance than the classical IMM algorithm in trajectory tracking, particularly in velocity estimation.

However, there are some shortcomings of the SIMM and MIMM, for example, the filter used in algorithms is a traditional linear Kalman filter which can only solve the linear tracking problem. But in practice, most of the target tracking is a multiple model nonlinear filtering problem, especially in polar coordinate sensing application [4]. In this paper, four improved IMM algorithms are proposed: EKF-SIMM, EKF-MIMM, UKF-SIMM and UKF-MIMM. The experiments confirm the efficiency of the proposed algorithms. And the UKF-MIMM algorithm has the best accuracy of estimation and the longest running time in all of the six algorithms. Although the EKF-SIMM has a higher error in position estimation, the computational complexity is the minimum.

2. The improved multiple model algorithms

The improved multiple model algorithms are scalar-weight IMM and matrix-weight IMM algorithm combined with UKF and EKF, named UKF-SIMM, UKF-MIMM, EKF-SIMM, EKF-MIMM.. The steps of algorithms are as follows:

2.1. The UKF-SIMM Algorithm

Step1: Calculating the mixed initial scalar-weight for the filter matched to model $M_j^k\left(j \in S = \{1, 2, \cdots s\}\right)$

$$P\left\{M_i^{k-1} \mid M_j^k, Z^{k-1}\right\} = \frac{\pi_{ij} a_i^{k-1}}{\bar{c}_j} \qquad i, j \in S \tag{1}$$

Where S is the set of models and the number of set elements is s, M_j^k denotes the model j at time k, π_{ij}=$prob\left\{M_j^k \mid M_j^{k-1}\right\} = P\left\{M_j^k \mid M_j^{k-1}\right\}$, a_i^{k-1}=$prob\left\{M_j^k \mid Z^{k-1}\right\} = P\left\{M_i^{k-1} \mid Z^{k-1}\right\}$, $\bar{c}_j = \sum_{i=1}^{m} \pi_{ij}\mu_i(k-1)$ is the normalization constant, $Z^{k-1} = \left[z(1), \cdots z(k-1)\right]$ is the sequence of measurement .

Step2: Calculating the mixed initial state $\hat{x}_{0j}\left(k \mid k\right)$ and corresponding covariance $P_{0j}\left(k \mid k\right)$ for the filter matched to model $M_j^k\left(j \in S\right)$

$$\hat{x}_{0j}\left(k\mid k\right)=\sum\nolimits_{i=1}^{s}a_{i|j}\left(k\mid k\right)\hat{x}_{j\_s\_ukf}^{k-1} \tag{2}$$

$$P_{0j}\left(k\mid k\right)=\sum\nolimits_{i=1}^{s}a_{i|j}\left(k\mid k\right)\{P_{i\_s\_ukf}^{k-1}+\left[\hat{x}_{j\_s\_ukf}^{k-1}-\hat{x}_{0j}\left(k\mid k\right)\right]\times\left[\hat{x}_{j\_s\_ukf}^{k-1}-\hat{x}_{0j}\left(k\mid k\right)\right]^{T}\} \tag{3}$$

Where $\hat{x}_{j\_s\_ukf}^{k-1}$ is the estimation of state based on the ith ukf at time $k-1$, and the corresponding covariance is $P_{i\_s\_ukf}^{k-1}$.

Step3: UKF Filtering produce outputs $\hat{x}_{j\_s\_ukf}^{k},P_{j\_s\_ukf}^{k}$ using algorithm [4].

Step4: Combining of the state estimations and corresponding covariance according to the updated scalar-weight.

$$\hat{x}_{s\_ukf}\left(k\right)=\sum\nolimits_{j=1}^{s}a_{j}^{k}\hat{x}_{j\_s\_ukf}^{k} \tag{4}$$

Updated scalar-weight of model M_{j}^{k} is

$$a_{j}^{k}=\left(\sum\nolimits_{i=1}^{s}\frac{1}{trP_{i\_s\_ukf}^{k}}\right)^{-1}\frac{1}{trP_{i\_s\_ukf}^{k}} \tag{5}$$

The error variance matrix of the optimal fusion estimation is

$$P_{s\_ukf}=\sum\nolimits_{j=1}^{s}\left(a_{j}^{k}\right)^{2}P_{i\_s\_ukf}^{k} \tag{6}$$

2.2. The EKF-SIMM Algorithm

The EKF-SIMM algorithm has the similar process with the UKF-SIMM algorithm except in step 3. That is, the estimation of state $\hat{x}_{j\_s\_ekf}^{k}$ and the corresponding covariance $P_{j\_s\_ekf}^{k}$ are calculated using the EKF Filtering produce outputs using the algorithm [5] in EKF-DIMM algorithm. Then the final combining outputs in step 4 are calculated as follows:

$$\hat{x}_{s\_ekf}\left(k\right)=\sum\nolimits_{j=1}^{s}a_{j}^{k}\hat{x}_{j\_s\_ekf}^{k}\quad,\quad a_{j}^{k}=\left(\sum\nolimits_{i=1}^{s}\frac{1}{trP_{i\_s\_ekf}^{k}}\right)^{-1}\frac{1}{trP_{i\_s\_ekf}^{k}}\quad,\quad P_{s\_ekf}=\sum\nolimits_{j=1}^{s}\left(a_{j}^{k}\right)^{2}P_{i\_s\_ekf}^{k} \tag{7}$$

2.3. The UKF-MIMM Algorithm

Step1: Calculating the mixed initial matrix-weight for the filter matched to model $M_{j}^{k}\left(j\in S\right)$

$$\Omega_{i|j}\left(k\mid k\right)=P\left\{M_{i}^{k-1}\mid M_{j}^{k},Z^{k}\right\}=\frac{\pi_{ij}\Omega_{i}^{k-1}}{\sum\nolimits_{i=1}^{s}\pi_{ij}\Omega_{i}^{k-1}}=\left(\sum\nolimits_{i=1}^{s}\pi_{ij}\Omega_{i}^{k-1}\right)^{-1}\pi_{ij}\Omega_{i}^{k-1} \tag{8}$$

Where $\Omega_{i}^{k-1}=prob\left\{M_{j}^{k}\mid Z_{j}^{k-1}\right\}=P\left\{M_{i}^{k-1}\mid Z^{k-1}\right\}$

Step2: Calculating the mixed initial state $\hat{x}_{0j}\left(k\mid k\right)$ and corresponding covariance $P_{0j}\left(k\mid k\right)$ for the filter matched to model $M_{j}^{k}\left(j=1,2\cdots s\right)$.

$$\hat{x}_{0j}\left(k\mid k\right)=\sum\nolimits_{i=1}^{s}\Omega_{i|j}\left(k\mid k\right)\hat{x}_{j\_m\_ukf}^{k-1} \tag{9}$$

$$P_{0j}\left(k\mid k\right)=\sum\nolimits_{i=1}^{s}\Omega_{i|j}\left(k\mid k\right)\{P_{i\_m\_ukf}^{k-1}+\left[\hat{x}_{j\_m\_ukf}^{k-1}\hat{x}_{0j}-\left(k\mid k\right)\right]\times\left[\hat{x}_{j\_m\_ukf}^{k-1}-\hat{x}_{0j}\left(k\mid k\right)\right]^{T}\} \tag{10}$$

Step3: UKF Filtering produce outputs $\hat{x}_{j\_m\_ukf}^{k},P_{j\_m\_ukf}^{k}$ using the algorithm [4]

Step4: Combining of the state estimations and corresponding covariance according to the updated matrix-weight

$$\hat{x}_{m\_ukf}(k) = \sum_{j=1}^{s} \Omega_j^k \hat{x}_{j\_m\_ukf}^k \tag{11}$$

Updated matrix-weight of model M_j^k is

$$\Omega_j^k = \left(\sum_{i=1}^{s} \left(P_{i\_m\_ukf}^k\right)^{-1}\right)^{-1} \left(P_{j\_m\_ukf}^k\right)^{-1} \tag{12}$$

And the error variance matrix of the optimal fusion estimator is

$$P_{m\_ukf} = \left(\left(\sum_{i=1}^{s} \left(P_{i\_m\_ukf}^k\right)^{-1}\right)^{-1}\right) \tag{13}$$

2.4. The EKF-MIMM Algorithm

The EKF-MIMM algorithm has the similar process with the UKF-MIMM algorithm except in step 3. That is, the estimation of state $\hat{x}_{j\_m\_ekf}^k$ and the corresponding covariance $P_{j\_m\_ekf}^k$ are calculated using the EKF Filtering produce outputs using the algorithm [5] in EKF-DIMM algorithm. Then the final combining outputs in step 4 are calculated as follows:

$$\hat{x}_{m\_ekf}(k) = \sum_{j=1}^{s} \Omega_j^k \hat{x}_{j\_m\_ekf}^k,\ \Omega_j^k = \left(\sum_{i=1}^{s} \left(P_{i\_m\_ekf}^k\right)^{-1}\right)^{-1} \left(P_{j\_m\_ekf}^k\right)^{-1},\ P_{m\_ekf} = \left(\left(\sum_{i=1}^{s} \left(P_{i\_m\_ekf}^k\right)^{-1}\right)^{-1}\right) \tag{14}$$

3. Simulation and comparison results

In this section, some experiments are conducted to evaluate the performance of EKF-SIMM, UKF-SIMM, EKF-MIMM and UKF-MIMM algorithm. Two goodness-of-fit are used to assess the tracking efficiency of the results: running time and Root Mean Square Error (RMSE). The running time can be calculated by the software and the RMSE is calculated as

$$RMSE_k = \sqrt{\frac{1}{M} \sum_{i=1}^{M} \left(\hat{x}_k^i - x_k^i\right)^2} \tag{15}$$

Where x_k^i and $\hat{x}_k^i$ denote the true and estimated state at the *ith* Monte Carlo run in time k, respectively, M is the total number of independent Monte Carlo runs.

The target trajectory is generated as follows: The target moves in different state for four periods. First, it moves with constant velocity $v_x = 0, v_y = -1$ in a straight line from 0s to 50s. Then it maneuvers and turns right with $\omega = \pi / 20$ from 51s to 70s. From 71s to 120s it moves with constant velocity $v_x = 1, v_y = 0$. For the last 30s, it maneuvers and turns right with $\omega = \pi / 20$. The position trajectory of the target is shown in Figure1, and the velocity trajectory of the target is shown in Figure2.

3.1. Simulation settings

For the tracking of the target, two models are employed: constant-turn (CT) model with constant angular rate and constant-acceleration (CA) model, the state representation for each model refers to [6]. The measurement model [6] in a polar form at discrete time, including range and bearing, is given by

$$z(k)=\begin{bmatrix} r_k \\ \phi_k \end{bmatrix}=h(x(k))+\upsilon_k=\begin{bmatrix} \sqrt{x_k^2+y_k^2} \\ \tan^{-1}\left(\dfrac{y_k}{x_k}\right) \end{bmatrix}+\begin{bmatrix} \upsilon_{rk} \\ \upsilon_{\phi k} \end{bmatrix} \tag{16}$$

Where r_k is polar radius, ϕ_k is the bearing value, υ_k is an additive zero-mean Gaussian noise vector with variance $R_k=diag\left\{\left[\sigma_r^2,\sigma_\phi^2\right]\right\}$, σ_ϕ^2 and σ_r^2 are standard deviations for range and bearing, respectively. The simulation parameters setting are as follows:

Sampling interval $T=1s$, the noise covariances are

$R_i=diag\left([\sigma_{ri}^2,\sigma_{\phi i}^2]\right)$ $\sigma_{r1}=\sigma_{r2}=\sqrt{5},\sigma_{\phi 1}=\sigma_{\phi 2}=1/5$

Initial state and error covariance are

$x(0)=\left[1,1,0,\text{-}1,0,0\right]'$ $P(0)=diag\left(\left[0.1,0.1,0.1,0.1,0.1,0.1\right]\right)$

The model transition matrix and initial distributions are

$\pi_{ij}=\begin{bmatrix} 0.95 & 0.05 \\ 0.05 & 0.95 \end{bmatrix}$ $\phi_1=0.9,\phi_2=0.1$ $a_1=0.9,a_1=0.1$

$\Omega_1=diag\left(\left[0.9,\ 0.8,\ 0.2,\ 0.5,\ 0.6,\ 0.1\right]\right)$ $\Omega_2=diag\left(\left[0.1,\ 0.2,\ 0.8,\ 0.5,\ 0.4,\ 0.9\right]\right)$

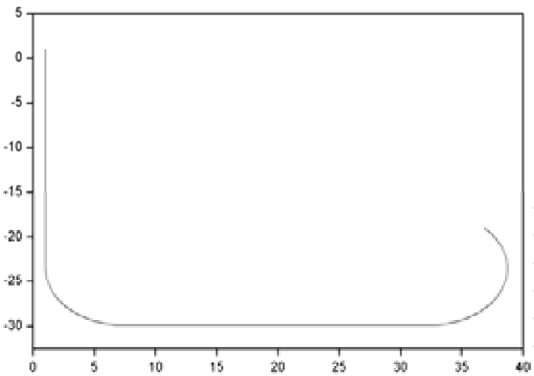

Figure 1.True position trajectory

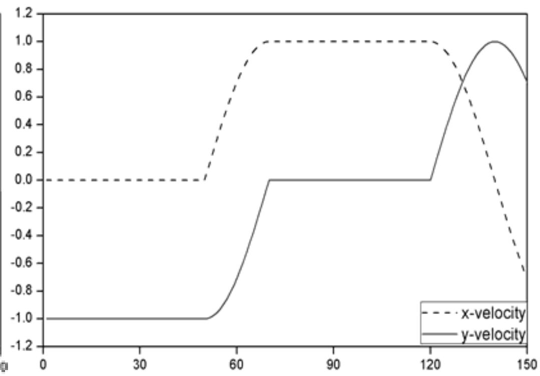

Figure 2. True Velocity trajectory

3.2. Results Analysis

The results are obtained from 100 Monte Carlo simulation runs. Figure3 shows the position tracking RMSE in x direction. From the figures, it can be obviously seen that the UKF-MIMM has the best accuracy performance and the EKF-SIMM algorithm performs relatively worse than the other five algorithms in position estimation. This may be caused by simplifying the process of calculating the mixed probability in EKF-SIMM. And it can also be seen that the Multiple Model algorithm combined with UKF is better than combined with EKF. It may be caused by the fact that the first-order EKF linearization can't handle the strong nonlinear system tracking well.

Figure4 shows the velocity tracking RMSE in x direction. It is can be seen that the velocity estimation accuracy of the proposed four algorithms is higher than EKF-IMM and UKF-IMM. And it can also be seen that the UKF-MIMM performs best and EKF- MIMM is only relatively poorer than the UKF-

MIMM in velocity estimation, which can be inferred that filters combined with MIMM algorithm have an absolute advantage in velocity estimation.

Table1 shows the comparisons of average RMSE and run-time statistics among the six algorithms. The running time is taken average after summing for 100times' simulation, each time 150steps.The

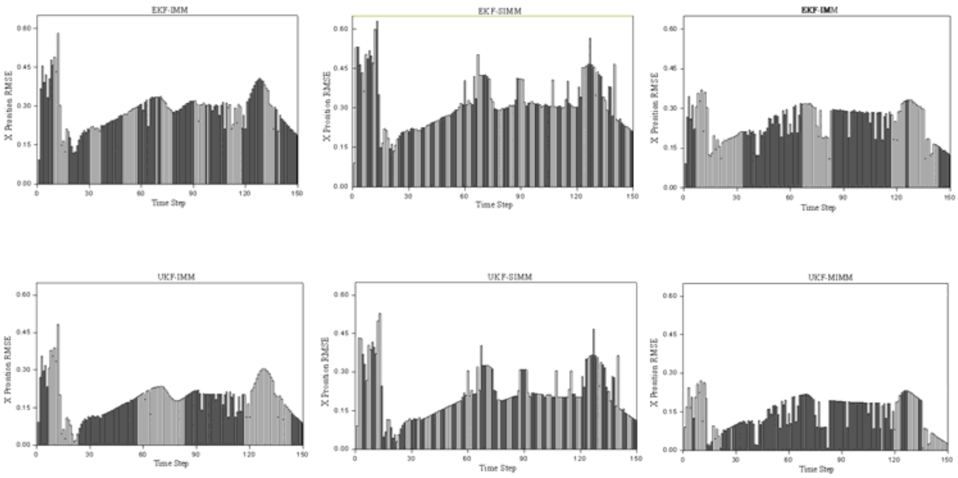

Figure3. The position tracking RMSE in x direction

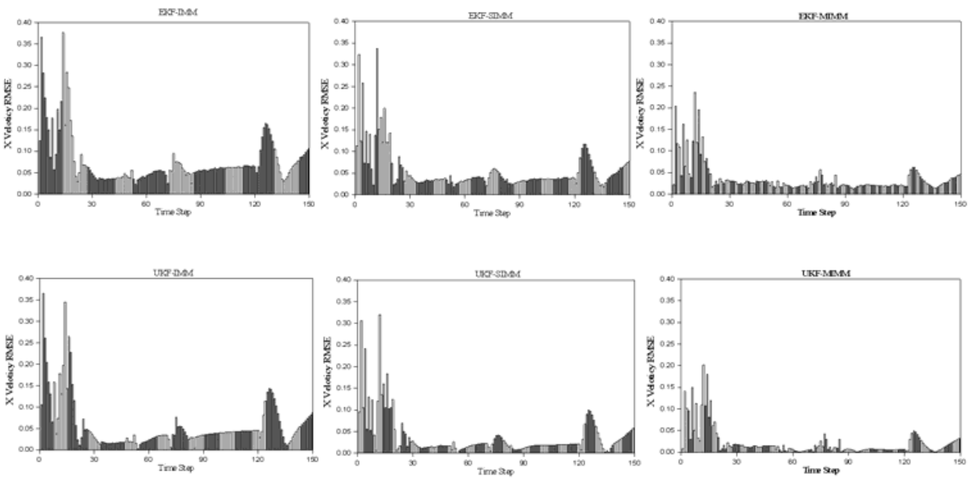

Figure4. The velocity tracking RMSE in x direction

simulation environment is MATLAB7.6.4 on a 2.93GHz 2CPU Intel core 2-based computer operating under Windows XP (Professional) system.

From the above simulation results, it can be seen that the UKF-SIMM, EKF-SIMM, EKF-MIMM and UKF-MIMM algorithm have an obvious advantage in velocity estimation. And the UKF-MIMM algorithm has the best accuracy of estimation and the longest running time in all of the six algorithms. Although the EKF-SIMM has a higher error in position estimation, the computational complexity is the minimum. Comprehensive evaluate the accuracy and time consuming performance index, EKF-MIMM may be a good choice in some velocity and position estimation scene.

4. Conclusions and future work

To solve the error problem that caused by reciprocal transformation probability density functions (PDFs) and probability masses in nonlinear IMM algorithm, four algorithms are proposed in this paper: UKF-SIMM, EKF-SIMM, UKF-MIMM and EKF-MIMM. The simulation results indicate that the EKF-SIMM, EKF-MIMM and UKF-MIMM algorithms have an obvious advantage in velocity estimation compared with EKF-IMM and UKF-IMM algorithm. Furthermore, the UKF-MIMM has the best accuracy of estimation and computational complexity while the EKF-SIMM operating time is the minimum. So, find an efficiency algorithm that can optimize computational complexity and accuracy is a meaningful work in the future.

Table 1. Comparisons of average RMSE and run-time statistics among the six algorithms

Filters	X Position	Y Position	X velocity	Y velocity	Run Time
EKF-IMM	0.2819	0.2708	0.0775	0.1478	0.0912
EKF-SIMM	0.3187	0.2985	0.0561	0.0964	**0.0601**
EKF-MIMM	0.2411	0.1977	0.0361	0.0517	0.1264
UKF-IMM	0.1827	0.1782	0.0576	0.1271	0.3394
UKF-SIMM	0.2194	0.1892	0.0379	0.0784	0.3087
UKF-MIMM	**0.1418**	**0.1056**	**0.0216**	**0.0332**	0.3829

Acknowledgements

This work is funded by Program for New Century Excellent Talents in University (Grant No.NCET-08-0333), Independent Innovation Foundation of Shandong University (Grant No.2010JC015) and Shandong Province Natural Science Foundation (Grant No. ZR2011FM039).

References

[1] Fu, X.; Jia, Y.; Du, J.; Yu, F. New interacting multiple model algorithms for the tracking of the manoeuvring target. IET Control Theory and Applications, October 2010; v 4, n 10, p 2184-2194.

[2] Johnston, L.A.; Krishnamurthy, V. An improvement to the interacting multiple model (IMM) algorithm. IEEE Transactions on Signal Processing, December 2001; v 49, n 12, p 2909-2923.

[3] Hong, L.; Cui, N.; Bakich, M.; Layne, J.R. MultiMate interacting multiple model particle filter for terrain-based ground target tracking. IEEE Proceedings: Control Theory and Applications, 2006; v 153, n 6, p 721-731.

[4] Cui, Ningzhou; Hong, Lang; Layne, Jeffery R. A comparison of nonlinear filtering approaches with an application to ground target tracking. Signal Processing, August 2005; v 85, n 8, p 1469-1492.

[5] Toledo-Moreo, Rafael; Zamora-Izquierdo, Miguel A.; Úbeda-Miñarro, Benito; Gómez-Skarmeta, Antonio F. High-Integrity IMM-EKF-Based Road Vehicle Navigation With Low-Cost GPS/SBAS/INS. IEEE Transactions on Intelligent Transportation Systems, September 2007; v 8, n 3, p 491-511.

Li, X. Rong; Jilkov, Vesselin P. Survey of Maneuvering Target Tracking. Part I: Dynamic Models. IEEE Transactions on Aerospace and Electronic Systems, October 2003; v 39, n 4, p 1333-1364.